

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 07-295816

(43)Date of publication of application : 10.11.1995

(51)Int.Cl. G06F 9/44

(21)Application number : 07-022546

(71)Applicant : INTERNATL BUSINESS MACH
CORP <IBM>

(22)Date of filing : 10.02.1995

(72)Inventor : CANTIN GUYLAINE
COPELAND GEORGE P
GHEITH AHMED M
SESSIONS ROGER H

(30)Priority

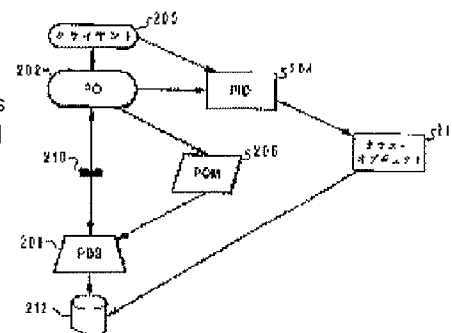
Priority number : 94 234435 Priority date : 26.04.1994 Priority country : US

(54) OPTIMIZING METHOD AND SYSTEM FOR ACCESS TO DATA STORAGE

(57)Abstract:

PURPOSE: To provide a method in a data processing system to optimize access to a data storage in an object-oriented environment.

CONSTITUTION: The data processing system includes an object and an identification object 204, and the identification object 204 is an instance of a class object 214. In the method, the identification object 204 receives a request to open a data storage 212. The request is fed from the identification object 204 to the class object 214. Whether or not the data storage 212 is opened is discriminated. When the data storage 212 is closed, the data storage 212 is opened and a message denoting it is fed to the identification object 204. If the data storage 212 is already opened, a message denoting it is fed to the identification object 204 to optimize the access to the data storage 212.



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平7-295816

(43) 公開日 平成7年(1995)11月10日

(51) Int.Cl.⁹

G 0 6 F 9/44

識別記号

5 3 0 S 7737-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数11 O L (全 13 頁)

(21) 出願番号 特願平7-22546

(22) 出願日 平成7年(1995)2月10日

(31) 優先権主張番号 2 3 4 4 3 5

(32) 優先日 1994年4月26日

(33) 優先権主張国 米国 (US)

(71) 出願人 390009531

インターナショナル・ビジネス・マシー
ズ・コーポレーション

INTERNATIONAL BUSIN
ESS MACHINES CORPO
RATION

アメリカ合衆国10504、ニューヨーク州
アーモンク (番地なし)

(72) 発明者 ガイレイン・カンティン

カナダM4W2B7、オンタリオ州、トロ
ント、ダグラス・ドライブ 190

(74) 代理人 弁理士 合田 潔 (外2名)

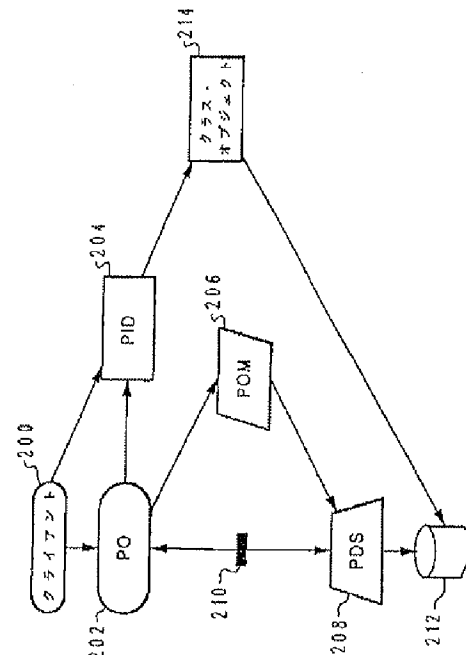
最終頁に続く

(54) 【発明の名称】 データストアへのアクセスの最適化方法及びシステム

(57) 【要約】

【目的】 オブジェクト指向環境においてデータストアへのアクセスを最適化するためのデータ処理システムにおける方法を提供する。

【構成】 データ処理システムが、オブジェクトと識別オブジェクトを含み、識別オブジェクトはクラス・オブジェクトのインスタンスである。その方法は、識別オブジェクトにてデータストアをオープンするべく要求を受取る。要求は識別オブジェクトからそのクラス・オブジェクトへ送られる。データストアがオープンしているか否かが判断される。データストアがクローズであればオープンされかつそのことを示すメッセージが識別オブジェクトへ送られる。もしデータストアが既にオープンであればそのことを示すメッセージが識別オブジェクトへ送られることによりデータストアへのアクセスが最適化される。



【特許請求の範囲】

【請求項1】オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するための方法であって、

(a) 前記識別オブジェクトにて前記データストアをオープンするための要求を受取るステップと、

(b) 前記オープン要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記オープン要求を送るステップと、

(c) 前記クラス・オブジェクトにて前記オープン要求を受取ったことに応答して、前記データストアがオープンしているか否かを判断するステップと、

(d) 前記データストアがクローズされていることに応答して、前記データストアをオープンしかつ前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るステップと、

(e) 前記データストアがオープンされているとの判断に応答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るステップとを含むデータ処理システムにおける方法。

【請求項2】(f) 前記識別オブジェクトにて前記データストアをクローズするための要求を受取るステップと、

(g) 前記識別オブジェクトにて前記クローズ要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記クローズ要求を送るステップと、

(h) 前記データストアをオープンしたままにする他のオブジェクト要求があるか否かを判断するステップと、

(i) 前記データストアをオープンしたままにする要求が無いとの判断に応答して、前記データストアをクローズするステップと、

(j) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送るステップとを含む請求項1に記載の方法。

【請求項3】オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するためのデータ処理システムにおいて、

(a) 前記識別オブジェクトにて前記データストアをオープンするための要求を受取る受信手段と、

(b) 前記オープン要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記オープン要求を送る第1の送信手段と、

(c) 前記クラス・オブジェクトにて前記オープン要求を受取ったことに応答して、前記データストアがオープンしているか否かを判断する判断手段と、

(d) 前記データストアがクローズされていることに応答して、前記データストアをオープンしかつ前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るオープン手段と、

(e) 前記データストアがオープンされているとの判断に応答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送る第2の送信手段とを有するデータ処理システム。

【請求項4】(f) 前記識別オブジェクトにて前記データストアをクローズするための要求を受取る第2の受信手段と、

(g) 前記識別オブジェクトにて前記クローズ要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記クローズ要求を送る第3の送信手段と、

(h) 前記データストアをオープンしたままにする他のオブジェクト要求があるか否かを判断する第2の判断手段と、

(i) 前記データストアをオープンしたままにする要求が無いとの判断に応答して、前記データストアをクローズするクローズ手段と、

(j) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送る第4の送信手段とを含む請求項3に記載のデータ処理システム。

【請求項5】前記データストアが、リレーショナル・データベースである請求項4に記載のデータ処理システム。

【請求項6】前記受信手段が、クライアント・オブジェクトから要求を受取る請求項5に記載のデータ処理システム。

【請求項7】永続オブジェクトをサポートするためのシステムであって、

(a) データストアへアクセスするための要求を要求オブジェクトから受取る受信手段と、

(b) 前記要求を受取ることに応答して、前記データストアへアクセスするアクセス手段とを有し、前記アクセス手段が、

(c) 前記データストアをオープンするための要求を受取ることに応答して、前記データストアがオープンしているか否かを判断するための第1の判断手段と、

(d) 前記データストアがクローズされているとの判断に応答して、前記データストアをオープンする第1のオープン手段と、

(e) 前記データストアが既にオープンされているとの判断に応答して、前記データストアを仮想的にオープンする第2のオープン手段と、

(f) 前記受信手段にて前記データストアをクローズするための要求を受取ったことに応答して、前記データストアをオープンしたままにする要求が存在するか否かを判断する第2の判断手段と、

(g) 前記データストアをオープンしたままにする要求が無いとの判断に回答して、前記データストアをクローズする第1のクローズ手段と、

(h) 前記データストアをオープンしたままにする要求があるとの判断に回答して、前記データストア仮想的にクローズする第2のクローズ手段とを有するシステム。

【請求項8】オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するために、前記データ処理システムが実行可能な命令をエンコードしかつ前記データ処理システムにより読取り可能なプログラム記憶装置であって、

(a) 前記識別オブジェクトにて前記データストアへアクセスするための要求を受取るための第1の命令手段と、

(b) 前記要求を受取ったことに回答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記要求を送るための第2の命令手段と、

(c) 前記クラス・オブジェクトにて前記データストアをオープンする要求を受取ったことに回答して、前記データストアがオープンしているか否かを判断するための第3の命令と、

(d) 前記データストアがクローズされていることに回答して、前記データストアをオープンしかつ前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るための第4の命令と、

(e) 前記データストアがオープンされているとの判断に回答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るための第5の命令と、

(f) 前記データストアをクローズするための要求を受取ったことに回答して、前記データストアをオープンしたままにするべく他のオブジェクトが要求しているか否かを判断するための第6の命令と、

(g) 前記データストアをオープンしたままにする要求が無いとの判断に回答して、前記データストアをクローズするための第7の命令と、

(h) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送るための第8の命令とを有するプログラム記憶装置。

【請求項9】前記プログラム記憶装置が、ハードディスク駆動装置である請求項8に記載のプログラム記憶装置。

【請求項10】前記プログラム記憶装置が、データ処理装置とともに使用するROMである請求項8に記載のプログラム記憶装置。

【請求項11】前記プログラム記憶装置が、データ処理装置とともに使用するフロッピー・ディスクである請求項8に記載のプログラム記憶装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、広くはオブジェクト・データのデータストアへアクセスする方法及びシステムに関し、特に、オブジェクト指向環境においてオブジェクト・データのデータストア(Datastore)のオープン

(開くこと)及びクローズ(閉じること)を最適化する方法及びシステムに関する。さらに、本発明は、オブジェクト指向環境におけるデータストアのオープン及びクローズに費やす時間を低減することによるオブジェクト指向環境におけるデータストアのオープン及びクローズのための方法及びシステムに関する。

【0002】

【従来の技術】オブジェクト指向プログラミング環境は、データのプレゼンテーション(表示)とその操作を、「オブジェクト」と呼ばれる単一のエンティティとして処理する。オブジェクトは、情報とその操作の記述とのパッケージである。オブジェクトは、そのオブジェクトに対して「メッセージ」を送ることによって操作される。メッセージを受取るオブジェクトは、自身をいかに操作するかを判断する。「クラス」は、1又は複数の関連するオブジェクトを記述し、そしてオブジェクトの「インスタンス」は、特定のクラスにより記述されたオブジェクトである。オブジェクトの各インスタンスは、「インスタンス変数」にデータを格納している。1つのクラスの中のオブジェクトの全てのインスタンスは、同じインスタンス変数を有しているが、そのインスタンス変数の実際の値は異なる場合もある。クラスは、「メソッド(方法:method)」を格納しており、これは、クラスのインスタンスに対してメッセージを送ることによって呼出される手順である。

【0003】インヘリタンス(継承:inheritance)

は、オブジェクト指向環境の特徴である。通常、オブジェクト指向環境は、クラス同士の間にインヘリタンスを与える。例えばクラスは、新しいクラスを形成するべく変更されることがあり、この場合元のクラスは「スーパークラス(superclass)」となる。そして新しいクラスは、「サブクラス(subclass)」と呼ばれ、そのスーパークラスから全てを引継いでいる。

【0004】オブジェクト指向環境の別の特徴は、オブジェクト永続性(persistence)である。オブジェクトは、「動的状態」と「永続状態」とを持つことができる。動的状態では、通常、オブジェクトがメモリ内にあり、そのオブジェクトの寿命全体に亘って存続することはない。例えば、動的状態のオブジェクトはシステム故障があった場合には残存しない。オブジェクトの永続状態とは、オブジェクトの動的状態を再生するために使用されるデータである。オブジェクト永続性を実現するための標準は、Object Management Group Inc.によって設定された。この標準は、Object Persistence Service S

pecification (OPSS) として知られており、オブジェクトの永続状態を維持しかつ管理するために用いられるインタフェースと機構とを与えるガイドラインを規定している。クライアントは、永続的オブジェクトを操作するオブジェクトである。

【0005】OPSSは、以下のような多数のインタフェースを規定する。すなわち、永続識別子(Persistent Identifier:PID)、永続オブジェクト(Persistent Object:PO)、永続オブジェクト・マネージャ(Persistent Object Manager:POM)、永続データ・サービス(Persistent Data Service:PDS)、プロトコル、及びデータストアである。PIDは、オブジェクトの永続状態の記憶場所を識別する情報を含む。POは、その永続性が通常その対象により外部から制御されるようなオブジェクトである。POは、そのPOの永続データを制御するオペレーションを規定する記憶/再記憶インタフェースを含む。PDSは、オブジェクトとデータストアとの間でデータを移動する。POSは、オブジェクトからの要求に関連する記憶をPDSへ経路指定する。プロトコルは、オブジェクトに入出するデータの移動のメカニズムを与える。「データストア」は、データを記憶するためのいくつかの方法のうちの1つを可能とするインタフェースである。「データストア」は、例えばデータベース、レコード・ファイル、又は他の形式のファイルであってもよく、またデータベース及びレコード・ファイルのための周知のインタフェースを用いてもよい。データストアは、オブジェクトの永続状態のための記憶を与える。

【0006】オブジェクト指向環境についてのさらに詳細な情報は、Peterson, Object Oriented Computing, IEEE Computer Society Press, IEEE Computer Society Press Order No. 821(1990) 及び De Champeaux, Object-Oriented System Development, Addison-Wesley Publishing Co. (1993)により得られる。オブジェクト永続性の実施に関するさらなる情報及び詳細は、Object Persistent Service Specification, OMG TC document number 93.11.3に記載があり、この文献は、49201d Connecticut Path, Framingham, Massachusetts, 01701, U. S. A.に所在するObject Management Group, Inc.から入手できる。

【0007】OPSSの下では、データストアへのアクセスに関して性能上の問題が存在する。データストアのオープンは、時間のかかるプロセスであるので、理想的にはできるだけ少ない頻度で行うべきである。通常、データストアがオープンされ、一連の記憶/再記憶オペレーションが実行され、その後にそのデータストアがクローズされる。OPSSの下では、オープン/クローズのオペレーションは、接続/切断のオペレーションの間に実行される。好ましくないことに、これらのオペレーションは、永続オブジェクト・インタフェースに規定され

ており、この中では、目標オブジェクトがインスタンス化される以前にはオペレーションを呼出すことができない。またそのオブジェクトが一旦非インスタンス化されると呼出すことができない。もしデータストアへアクセスしているオブジェクトが、ループ内でインスタンス化されたり非インスタンス化されたりするならば、そのデータストアはそのループが発生する毎にオープンされまたクローズされなければならない、性能を低下させる。従って、データストアのオープン及びクローズを最適化するための方法及びシステムを実現することは有用である。

【0008】

【発明が解決しようとする課題】本発明の目的は、データストアへアクセスするための方法及びシステムを提供することである。さらに、本発明の目的は、オブジェクト指向環境においてデータストアのオープン及びクローズを最適化するための方法及びシステムを提供することである。さらに、本発明の目的は、オブジェクト指向環境においてデータストアのオープン及びクローズのために費やされる時間を低減することによって、オブジェクト指向環境におけるデータストアのオープン及びクローズを最適化するための方法及びシステムを提供することである。

【0009】

【課題を解決するための手段】上記の目的は、以降に記載のとおり達成される。本発明は、オブジェクト指向環境においてデータストアへのアクセスを最適化するための、データ処理システムにおける方法を提供する。このデータ処理システムはオブジェクトと識別オブジェクトとを含み、そして、識別オブジェクトはクラス・オブジェクト内にある。この方法は、データストアをオープンするための要求を識別オブジェクトにおいて受信することを含む。この要求は、その受信に回答して識別オブジェクトからクラス・オブジェクトへ送られる。その後、クラス・オブジェクトにおける受信に回答してそのデータストアがオープンしているか否かが判断される。データストアがクローズしている場合にはそのデータストアがオープンされ、かつデータストアがオープンされたことを示すメッセージが識別オブジェクトへ送られる。データストアが既にオープンされている場合は、データストアがオープンされていることを示すメッセージが識別オブジェクトへ送られる。このようにして、データストアへのアクセスが最適化される。

【0010】データストアをクローズする際には、データストアをクローズするための要求を識別オブジェクトにおいて受信する。この要求は、その受信に回答して識別オブジェクトからクラス・オブジェクトへ送られる。他のオブジェクトがデータストアをオープンしたままにするよう要求しているか否かが判断される。データストアをオープンしたままにする要求が無いことに応答し

て、データストアがクローズされる。データストアがクローズされたことを示すメッセージが識別オブジェクトへ送られる。PIDのクライアントは、このいずれも認識しない。クライアントにとっては、データストアのオープンやクローズが、あたかもオープン要求やクローズ要求に回答して行われたかのように見える。本発明の上記の目的、特徴及び有用性は、以降の記載により明らかにされるであろう。

【0011】

【実施例】図1は、本発明を利用できるデータ処理システムであるパーソナル・コンピュータ・システム10を示している。図示のとおり、パーソナル・コンピュータ・システム10は、多くの構成要素からなり、それが互いに接続されている。特に、システム・ユニット12は、任意のモニタ14（例えば、汎用的ビデオ・ディスプレイ）に接続されかつこれを駆動することができる。さらに、システム・ユニット12は、任意にPCキーボード16またはマウス18等の入力装置を接続することができる。マウス18は、左右のボタン（図示せず）を有する。左ボタンは、一般的に主選択ボタンとして用いられ、別の言い方では、第1マウス・ボタンまたはマウス・ボタン1と呼ばれている。右ボタンは、通常、後述の補助的な機能を選択するために用いられる。右マウス・ボタンは、別の言い方では、第2マウス・ボタンまたはマウス・ボタン2と呼ばれている。さらに、プリンタ20等の任意の出力装置もシステム・ユニット12に接続することができる。最後に、システム・ユニット12は、ディスク駆動装置22等の1または複数の大容量記憶装置を備えていてもよい。

【0012】以下に記載するように、システム・ユニット12は、PCキーボード16、マウス18、またはローカル・エリア・ネットワーク用インタフェース等の入力装置に対して応答する。さらに、フロッピー・ディスク駆動装置22、ディスプレイ14、プリンタ20、及びローカル・エリア・ネットワーク通信システム等の入出力（I/O）装置が、周知の方法でシステム・ユニット12に接続されている。もちろん、当業者であれば、他の汎用的な構成要素をもまたシステム・ユニット12に接続し相互に作用させることができることは自明であろう。本発明によれば、パーソナル・コンピュータ・システム10はシステム・プロセッサを有しており、これはランダム・アクセス・メモリ（RAM）、読取り専用メモリ（ROM）、及び複数のI/O装置と相互に接続されている。

【0013】通常の使用においては、パーソナル・コンピュータ・システム10は、サーバとして小グループのユーザに対してあるいは単一のユーザに対して独立した演算能力を提供するために設計することができ、よって個人や小企業による購入が可能であるような低価格とされている。オペレーションにおいては、システム・プロ

セッサは、IBMのOS/2オペレーティング・システムまたはDOS等のオペレーティング・システムの下で機能する。「OS/2」は、International Business Machines Corporationの米国登録商標である。このタイプのオペレーティング・システムは、I/O装置とオペレーティング・システムとの間に基本入出力システム（BIOS:Basic Input/Output System）インタフェースを備えている。BIOSは、マザーボードすなわちプレーナ上のROMに記憶されており、POSTと称されるパワーオン自己検査に含まれる診断ルーチンを含んでいる。

【0014】本発明による上記の構造について説明するに先立って、パーソナル・コンピュータ・システム10の一般的なオペレーションの概要に触れた方がよいであろう。図2は、本発明によるパーソナル・コンピュータ・システム10の種々の構成要素を示したブロック図である。図2はさらに、プレーナ11の構成要素及びI/Oスロット46aから46dへのプレーナ11の接続並びにパーソナル・コンピュータ・システム10の他のハードウェアを示している。プレーナ11へ接続されているのは、マイクロプロセッサからなる中央演算処理装置（CPU）26であり、高速CPUローカル・バス24によりバス制御タイミング装置38を介してメモリ制御装置50へ接続されている。メモリ制御装置50はさらに、揮発性ランダム・アクセス・メモリ（RAM）58へ接続されている。任意の適切なマイクロプロセッサをCPU26として使用できるが、一つの好適なマイクロプロセッサとして、Intel Corporationから販売されているペンティアム（Pentium）・マイクロプロセッサがある。「Pentium」は、Intel Corporationの米国商標である。

【0015】本発明について、特に図2のシステム・ブロック図を参照して以降に記述するが、記述するにあたって先ず理解されたい点は、本発明による装置及び方法は他のハードウェア構成を持つプレーナ・ボードにおいても適用することができることである。例えば、システム・プロセッサとしては、Intel 80286、80386、または80486のマイクロプロセッサがある。これらの特定のマイクロプロセッサは、リアル・アドレッシング・モード(real addressing mode)においてもプロテクト・アドレッシング・モード(protect addressing mode)においても動作することができる。各モードは、マイクロプロセッサのメモリの異なる領域にアクセスするためのアドレス指定方式を提供する。

【0016】図2を参照すると、CPUローカル・バス24（データ、アドレス、及び制御の各要素からなる）は、CPU26、任意の数値演算コプロセッサ27、キャッシュ制御装置28、及びキャッシュ・メモリ30を接続している。CPUローカル・バス24にはさらに、バッファ32が接続されている。バッファ32自体は、

(CPUローカル・バスに比べて)より低速のシステム・バス34へ接続されており、システム・バス34もまたアドレス、データ、及び制御の各要素からなる。システム・バス34は、バッファ32と更なるバッファ36との間を接続している。さらにシステム・バス34は、バス制御タイミング装置38及び直接メモリ・アクセス(DMA)装置40へ接続されている。DMA装置40は、中央調停装置48とDMA制御装置41とからなる。バッファ36は、システム・バス34と例えばマイクロ・チャンネル・バス44等の任意の特性のバスとの間のインタフェースを行う。「Micro Channel」は、International Business Machines Corporationの米国登録商標である。マイクロ・チャンネル・バス44には、複数のI/Oスロット46aから46dが接続され、マイクロ・チャンネル・アダプタ・カードを受容する。これらのカードはさらに、I/O装置やメモリを接続することができる。図示の例では、I/Oスロット46cは、ハード・ディスク駆動装置を接続している。I/Oスロット46bは、CD-ROM駆動装置を接続している。そしてI/Oスロット46aは、ROMを搭載したアダプタ・カードを接続している。調停制御バス42は、DMA制御装置41及び中央調停装置48をI/Oスロット46及びディスク・アダプタ82へ接続している。システム・バス・バス34にはさらにメモリ制御装置50が接続されており、このメモリ制御装置50は、メモリ制御部52、アドレス・マルチプレクサ54、及びデータ・バッファ56からなる。メモリ制御装置50はさらに、RAMモジュール58で表現されるランダム・アクセス・メモリへ接続される。メモリ制御部52は、CPU26に対してRAM58の特定の領域へアドレスをマッピングするための論理を含んでいる。パーソナル・コンピュータ・システム10は、基本の1メガバイトのRAMモジュールとともに示されているが、図2において任意のメモリ・モジュール60から64として表現されている付加的なメモリを接続することができるものと理解されたい。

【0017】さらに、システム・バス34とプレーナI/Oバス68との間にバッファ66が接続されている。プレーナI/Oバス68は、アドレス、データ、及び制御の各要素を備えている。プレーナI/Oバス68には、種々のI/Oアダプタ及びディスプレイ・アダプタ70(これは任意のディスプレイ14を駆動するために用いられる)、クロック72、不揮発性RAM74(以降、「NVRAM」と称する)、RS232アダプタ76、パラレル・アダプタ78、複数のタイマ80、ディスク・アダプタ82、PCキーボード/マウス制御装置84、及び読取り専用メモリ(ROM)86等の他の周辺機器を接続している。ROM86は、多くのI/O装置間でユーザには見えない通信を可能とするBIO Sを含んでいる。

【0018】クロック72は、日付計算のために用いられる。NVRAM74は、システム・コンフィギュレーション(構成)・データを記憶するために用いられる。すなわち、NVRAMは、システムの現在のコンフィギュレーションを記述する値を格納することになる。例えば、NVRAM74は、固定ディスクまたはディスクットの容量、ディスプレイの形式、メモリの容量等を記述する情報を格納する。特に重要な点は、NVRAM74が、システム・コンソール(操作卓)・コンフィギュレーションを記述するために用いられるデータを格納することである。すなわち、PCキーボードがキーボード/マウス制御装置84へ接続されているか否か、ディスプレイ制御装置は使用可能であるか否か、あるいはASCIIターミナルがRS232アダプタ76へ接続されているか否か等のデータである。さらに、これらのデータは、特定のコンフィギュレーション(構成)・プログラムが実行されるときは必ずNVRAM74に記憶される。コンフィギュレーション・プログラムの目的は、システムのコンフィギュレーションを特定する値をNVRAM76へ記憶し、システムの電源が落ちるまで保存することである。

【0019】キーボード/マウス制御装置84には、ポートA及びBが接続されている。これらのポートは、(ASCIIターミナルではない)PCキーボード及びマウスをPCシステムへ接続するために用いられる。RS232アダプタ装置76には、RS232コネクタが接続される。このコネクタを介して任意のASCIIターミナルをシステムへ接続することができる。

【0020】特に、パーソナル・コンピュータ・システム10は、例えばIBMのPS/2やIBMのRISC SYSTEM/6000等の任意の適切なコンピュータを利用して実現することができる。上記の製品は、New York, Armonkに所在するInternational Business Machines Corporationの製品である。「RISC SYSTEM/6000」は、International Business Machines Corporationの米国商標であり、「PS/2」は、International Business Machines Corporationの米国登録商標である。

【0021】オブジェクトは、データ及びそのデータに関して操作するために必要な方法を格納する。オブジェクトは、本発明の好適例による図3に示されるような「ドーナツ図」によって表現することができる。オブジェクト・データ102は、適用可能な方法(メソッド)104乃至114により囲まれるオブジェクト100の中心に描かれている。データ102は、このオブジェクトの方法によってのみ変更可能である。方法104乃至114は、他のオブジェクトからのメッセージを受取ることによって呼出される。通常のオブジェクト指向システムは、メッセージ・ルータ(router)116を備えており、オブジェクト間でメッセージの経路指定を行う。よ

って、オブジェクト118がメッセージ・ルータ116へメッセージを送ることによって方法108を呼出し、続いてメッセージ・ルータ116がオブジェクト100の方法108に対してそのメッセージを送る。オブジェクト100は、外部のデータストア（図示せず）に記憶されたデータを有する永続オブジェクトの場合もある。

【0022】図4は、オブジェクト指向システムの永続オブジェクトをサポートするために用いられる構成要素を示している。クライアント200は、永続オブジェクト（PO）202を操作する。PO202は、関連する永続識別子（PID）204を有する。永続オブジェクト・マネージャ（POM）206は、PO202の永続的オペレーションのためのインタフェースを行う。永続データ・サービス（PDS）208は、POM206から要求を受取り、PO202からプロトコル210を通してデータを受取る。PDS208は、データストア212へのインタフェースを行い、データストア212はPO202のデータを記憶する。

【0023】先ずクライアントは、データストア212をオープンするためにPO202へ要求を送る。これに
20 応答して、PO202はデータストア212をオープンするためにPOM206に対して要求を送る。続いてPOM206は、PDS208を用いてデータストア212をオープンする。データストア212がオープンされたならば、PO202からのデータが、プロトコル210を用いてPDS208へ送られる。それからPDS208は、データストア212にデータを記憶する。最後に、データストア212がクローズされる。

【0024】次に図5は、オブジェクト永続性サポートにおける構成要素のブロック図及びそれらの相互作用を、本発明の好適例について示している。PID204は、クラス・オブジェクト214のインスタンスである。本発明においては、クライアント200は、データストア212をオープンするための要求を、PO202ではなくPID204へ送る。PID204は、この要求をクラス・オブジェクト214へ送り、クラス・オブジェクト214は、データストア212をオープンするための方法を含む。図示の例では、クラス・オブジェクト214が直接データストア212をオープンする。データストア212は、PDS208に対して要求することを通して、または、クラス・オブジェクト214によってもオープンすることができる。PID204は、本発明の好適例によってデータストアをオープンしかつクローズするためのオープン命令及びクローズ命令を含む。PID204は、クライアント200からのオープン要求及びクローズ要求をクラス・オブジェクト214へ渡すことによってデータストア212へアクセスするので、オープン及びクローズは、仮想的な場合と、クラス・オブジェクト214により処理される物理的なオープンやクローズを伴う場合とがある。これらの態様は、
30 40 50

クライアント200の認知からは隠されている。「データストア(Datastore)」は、データが記憶される場所を称する一般的な用語である。例えば、データストアは、関係データベース、オブジェクト指向データベース、レコード・ファイル、またはテープ駆動装置である。

【0025】図6は、クラス・オブジェクトを用いてデータストアをオープンするプロセスの流れ図を示している。ブロック300において、クラス・オブジェクトはPIDからデータストアをオープンするための要求を受取る。ブロック302において、データストアをオープンする要求を受取ったことに応答して、データストアがオープンしているか否かを判断する。もしデータストアがオープンされていなければ、ブロック304においてデータストアがオープンされる。このステップはデータストアの実際のすなわち物理的なオープンを表している。変数COUNTは、ブロック306においてゼロにセットされる。その後、ブロック308において、COUNTが1だけ増分される。そして、ブロック310において、
40 クラス・オブジェクトは、データストアがオープンされていることを示すコードをPIDに対して返す。この場合、データストアは物理的にオープンされている。

【0026】ブロック302に戻って、データストアが既にオープンされている場合は、このプロセスは直接ブロック308へ進み、COUNTが1だけ増分される。この場合は、プロセスは「仮想的に」データストアをオープンし、これはCOUNTにより追跡される。ブロック10において、クラス・オブジェクトは、データストアがオープンされていることを示すコードをPIDに対して返す。

【0027】図7は、データストアをクローズするプロセスの流れ図を、本発明の好適例について示している。ブロック320において、クラス・オブジェクトはPIDからデータストアをクローズするための要求を受取る。ブロック322において、要求を受取ったことに応答してクラス・オブジェクトはCOUNTを1だけ減分する。その後、ブロック324において、COUNTがゼロに等しいか否かを判断する。COUNTがゼロでないならば、
40 クラス・オブジェクトは、データストアがクローズされていることを示すコードをPIDに対して返す。この場合、データストアは仮想的にクローズされている。

【0028】ブロック324において、COUNTがゼロであるならば、ブロック328においてクラス・オブジェクトはデータストアをクローズする。データストアは物理的にクローズされる。その後、クラス・オブジェクトはデータストアがクローズされていることを示すコードをPIDに対して返す。クライアント・オブジェクトは、データストアがクローズされていることをPIDにより告げられるが、これは本発明にはよらない。

【0029】図5乃至図7に示したプロセスは、当業者であれば図1及び図2に示したデータ処理システムにお

いて実施可能である。本発明によるプロセスはさらに、データ処理システムにより読取り可能なプログラム記憶装置において実施可能であり、この場合、プログラム記憶装置は、本発明のプロセスをコード化したデータ処理システムが実行可能な命令をエンコードする。プログラム記憶装置は、例えば、当業者には周知のハードディスク駆動装置、フロッピーディスク駆動装置、光学ディスク駆動装置、ROM、及びEPROM等を含む様々な形態をとることができるが、これらに例に限定されるものではない。プログラム記憶装置に関するプロセスは、データ処理システムによりそのプログラム記憶装置を用いることによって稼働されるまでは、休止状態である。例えば、本発明によるデータ処理システムが実行可能な命令を格納するハードディスク駆動装置を、データ処理システムに接続してもよい。あるいは、本発明によるデータ処理システムが実行可能な命令を格納するフロッピーディスクを、データ処理システムのフロッピーディスク駆動装置に挿入してもよい。あるいは、本発明によるデータ処理システムが実行可能な命令を格納するROMを、I/Oスロットへ接続されるカードまたはアダプタを介してデータ処理システムに接続してもよい。

【0030】まとめとして、本発明の構成に関して以下の事項を開示する。

【0031】(1) オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するための方法であって、(a) 前記識別オブジェクトにて前記データストアをオープンするための要求を受取るステップと、(b) 前記オープン要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記オープン要求を送るステップと、(c) 前記クラス・オブジェクトにて前記オープン要求を受取ったことに応答して、前記データストアがオープンしているか否かを判断するステップと、(d) 前記データストアがクローズされていることに応答して、前記データストアをオープンしかつ前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るステップと、(e) 前記データストアがオープンされているとの判断に応答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るステップとを含むデータ処理システムにおける方法。

(2) (1) 前記識別オブジェクトにて前記データストアをクローズするための要求を受取るステップと、

(g) 前記識別オブジェクトにて前記クローズ要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記クローズ要求を送るステップと、(h) 前記データストアをオープンしたままにする他のオブジェクト要求があるか否かを判断するステッ

プと、(i) 前記データストアをオープンしたままにする要求が無いとの判断に応答して、前記データストアをクローズするステップと、(j) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送るステップとを含む上記(1)に記載の方法。

(3) オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するためのデータ処理システムにおいて、(a) 前記識別オブジェクトにて前記データストアをオープンするための要求を受取る受信手段と、(b) 前記オープン要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記オープン要求を送る第1の送信手段と、

(c) 前記クラス・オブジェクトにて前記オープン要求を受取ったことに応答して、前記データストアがオープンしているか否かを判断する判断手段と、(d) 前記データストアがクローズされていることに応答して、前記データストアをオープンしかつ前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るオープン手段と、(e) 前記データストアがオープンされているとの判断に応答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送る第2の送信手段とを有するデータ処理システム。

(4) (f) 前記識別オブジェクトにて前記データストアをクローズするための要求を受取る第2の受信手段と、(g) 前記識別オブジェクトにて前記クローズ要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記クローズ要求を送る第3の送信手段と、(h) 前記データストアをオープンしたままにする他のオブジェクト要求があるか否かを判断する第2の判断手段と、(i) 前記データストアをオープンしたままにする要求が無いとの判断に応答して、前記データストアをクローズするクローズ手段と、(j) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送る第4の送信手段とを含む上記(3)に記載のデータ処理システム。

(5) 前記データストアが、リレーショナル・データベースである上記(4)に記載のデータ処理システム。

(6) 前記受信手段が、クライアント・オブジェクトから要求を受取る上記(5)に記載のデータ処理システム。

(7) 永続オブジェクトをサポートするためのシステムであって、(a) データストアへアクセスするための要求を要求オブジェクトから受取る受信手段と、(b) 前記要求を受取ることに応答して、前記データストアへアクセスするアクセス手段とを有し、前記アクセス手段が、(c) 前記データストアをオープンするための要求

を受取ることに応答して、前記データストアがオープンしているか否かを判断するための第1の判断手段と、

(d) 前記データストアがクローズされているとの判断に応答して、前記データストアをオープンする第1のオープン手段と、(e) 前記データストアが既にオープンされているとの判断に応答して、前記データストアを仮想的にオープンする第2のオープン手段と、(f) 前記受信手段にて前記データストアをクローズするための要求を受取ったことに応答して、前記データストアをオープンしたままにする要求が存在するか否かを判断する第2の判断手段と、(g) 前記データストアをオープンしたままにする要求が無いとの判断に応答して、前記データストアをクローズする第1のクローズ手段と、(h) 前記データストアをオープンしたままにする要求があるとの判断に応答して、前記データストア仮想的にクローズする第2のクローズ手段とを有す

るシステム。(8) オブジェクト指向環境にて、オブジェクトと識別オブジェクトとを含みかつ前記識別オブジェクトがクラス・オブジェクト内にあるデータ処理システムにおけるデータストアへのアクセスを最適化するために、前記データ処理システムが実行可能な命令をエンコードしかつ前記データ処理システムにより読取り可能なプログラム記憶装置であつて、(a) 前記識別オブジェクトにて前記データストアへアクセスするための要求を受取るための第1の命令手段と、(b) 前記要求を受取ったことに応答して、前記識別オブジェクトから前記クラス・オブジェクトへ前記要求を送るための第2の命令手段と、(c) 前記クラス・オブジェクトにて前記データストアをオープンする要求を受取ったことに応答して、前記データストアがオープンしているか否かを判断するための第3の命令と、(d) 前記データストアがクローズされていることに応答して、前記データストアをオープンしかつ前記データストアがオープンされている

```
main0
{
    PID pid;
    PIDFactory pidfact;
    Account acctnt;
    boolean done;
    string acctntNo;

    pidfact = PIDFactoryNew0
    pid = create_PID_from_key(pidfact, "PID_DB2")
    acctnt_AccountNew0;
    __set_dbalias(pid, "SAMPLE");
    __set_userid(pid, "Sherlock Holmes")
    __set_password(pid, "Pipe")
    __open_datastore(pid);

    done = FALSE;
```

ことを示すメッセージを前記識別オブジェクトへ送るための第4の命令と、(e) 前記データストアがオープンされているとの判断に応答して、前記データストアがオープンされていることを示すメッセージを前記識別オブジェクトへ送るための第5の命令と、(f) 前記データストアをクローズするための要求を受取ったことに応答して、前記データストアをオープンのままにするべく他のオブジェクトが要求しているか否かを判断するための第6の命令と、(g) 前記データストアをオープンのままにする要求が無いとの判断に応答して、前記データストアをクローズするための第7の命令と、(h) 前記データストアがクローズされていることを示すメッセージを前記識別オブジェクトへ送るための第8の命令とを有するプログラム記憶装置。

(9) 前記プログラム記憶装置が、ハードディスク駆動装置である上記(8)に記載のプログラム記憶装置。

(10) 前記プログラム記憶装置が、データ処理装置とともに使用するROMである上記(8)に記載のプログラム記憶装置。

(11) 前記プログラム記憶装置が、データ処理装置とともに使用するフロッピー・ディスクットである上記(8)に記載のプログラム記憶装置。

【0032】

【発明の効果】本発明の有用性は、PIDによるオープン及びクローズが、仮想的である場合と、クラス・オブジェクトにより処理されるデータストアへの物理的接続を伴う場合とがあることである。よって、クライアント・オブジェクトについての性能が向上する。表1は、本発明によりデータストアをクローズ及びオープンするためにPIDを用いるクライアント・オブジェクトの実施態様を示している。

【0033】

【表1】

```

while (!done) {
    acctNo = readAcctNumber0;
    _put_string_item(pid, acctNo);
    _restore(acct, pid);
    updateAccount(acct);
    _store(acct, pid);
};
/*...*/
_close_datastore(pid);
}

```

【図面の簡単な説明】

【図1】 本発明を実施可能なパーソナル・コンピュータの態様によるデータ処理システムを示す図である。

【図2】 本発明によるパーソナル・コンピュータ・システムの様々な構成要素を示したブロック図である。

【図3】 オブジェクト指向環境におけるオブジェクトの構成図である。

【図4】 オブジェクト指向環境における永続オブジェクトをサポートするために用いられる構成要素を示した図である。

【図5】 本発明の好適例における永続オブジェクトをサポートするための構成要素及びそれらの関係を示した図である。

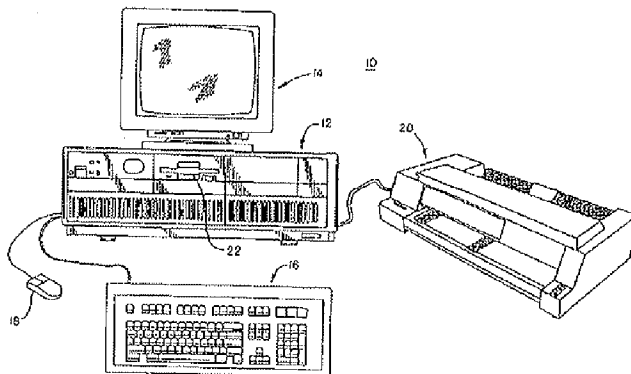
【図6】 クラス・オブジェクトを用いたデータストアをオープンするためのプロセスの流れ図である。

【図7】 本発明の好適例におけるデータストアをクローズするためのプロセスの流れ図である。

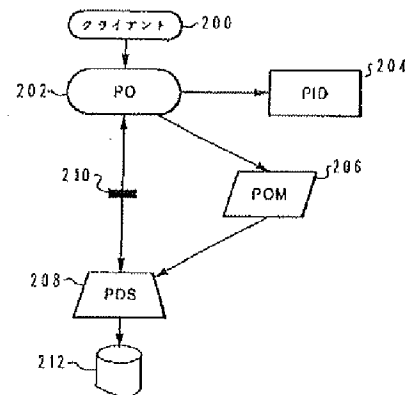
【符号の説明】

- 10 パーソナル・コンピュータ・システム
- 11 プレーナ
- 24 高速CPUローカル・バス
- 26 中央演算処理装置
- 34 システム・バス
- 38 バス制御タイミング装置
- 40 直接メモリ・アクセス装置
- 46 a、46 b、46 c、46 d I/Oスロット
- 50 メモリ制御装置
- 58 揮発性ランダム・アクセス・メモリ (RAM)
- 68 プレーナI/Oバス

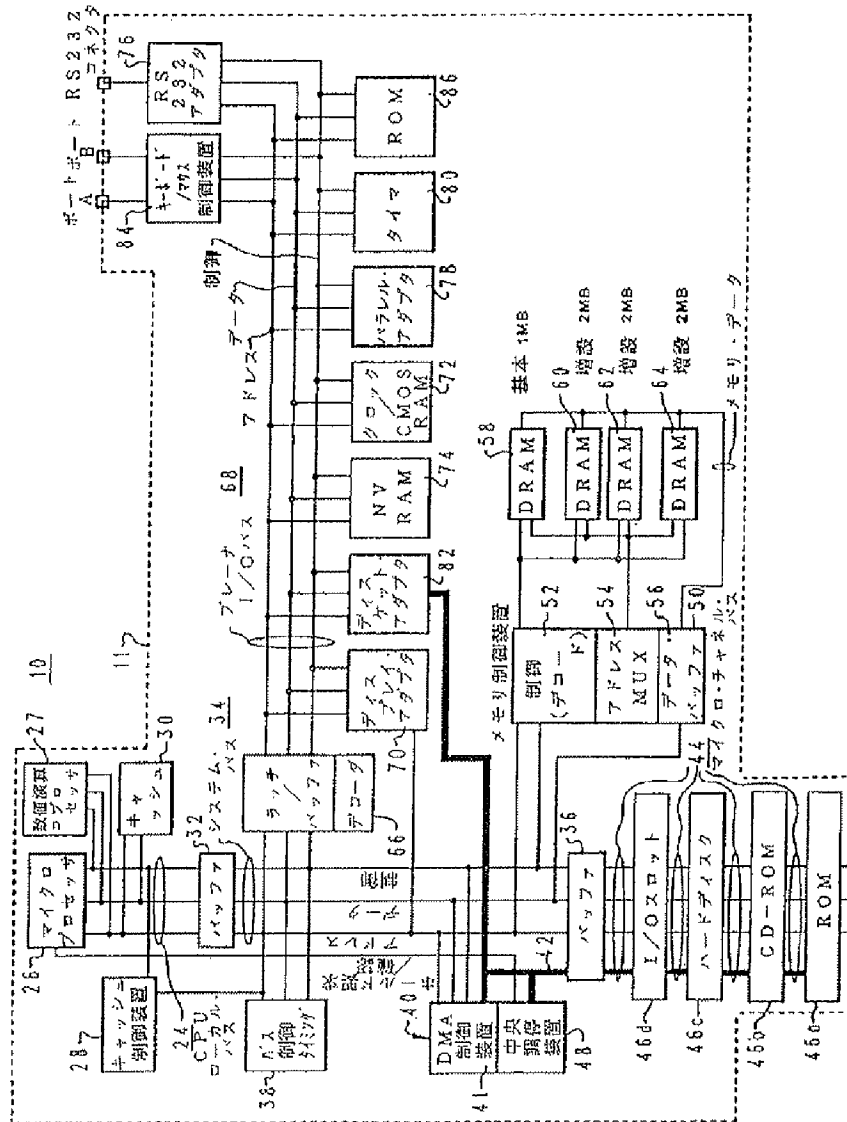
【図1】



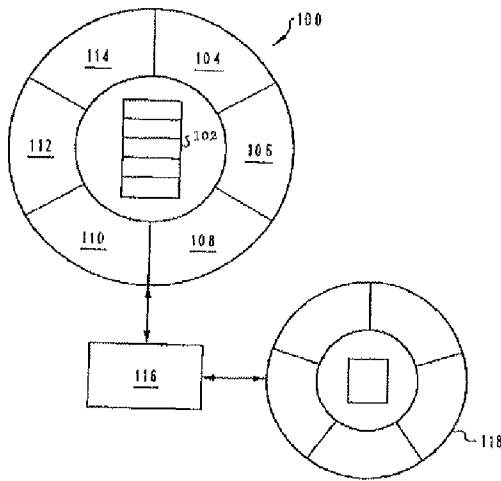
【図4】



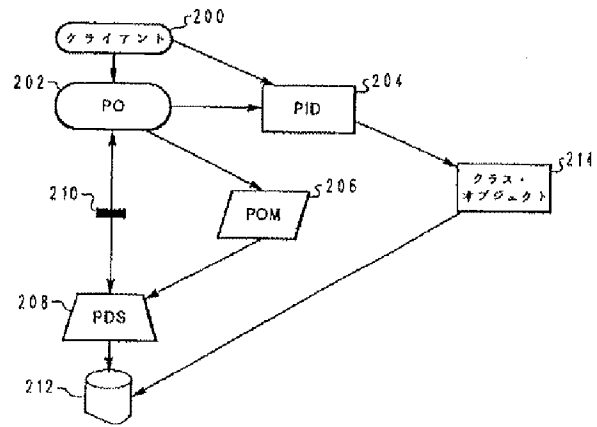
【図2】



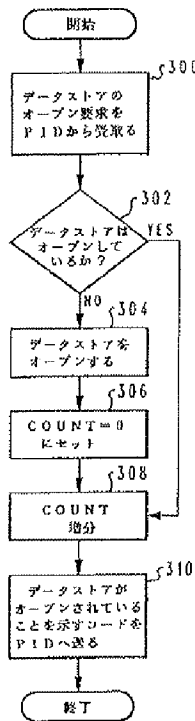
【図 3】



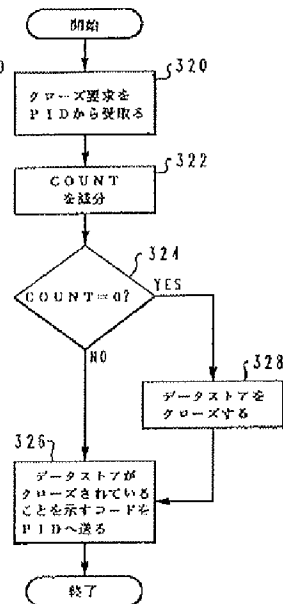
【図 5】



【図 6】



【図 7】



フロントページの続き

(72) 発明者 ジョージ・ビー・コーブランド
アメリカ合衆国78746 テキサス州、オー
スチン、メイル・スプリングス・ドライブ
1708

(72) 発明者 アーメッド・エム・ゲイス
アメリカ合衆国78681 テキサス州、ラウ
ンド・ロック、パロット・トレイル 1212
(72) 発明者 ロジャー・エイチ・セッションズ
アメリカ合衆国78750 テキサス州、オー
スチン、ペンスウッド・ドライブ 11414

